AUTOMATING ARGUMENT CONSTRUCTION

Technical Note No. 416

November 1986

By: John D. Lowrance
      Artificial Intelligence Center
      Computer Science and Technology Division

APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED

Automating Argument Construction

John D. Lowrance

Artificial Intelligence Center
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

Abstract

Over the past five years the Artificial Intelligence Center at SRI has been developing a new technology to address the problem of automated information management within real-world contexts. The result of this work is a body of techniques for automated reasoning from evidence that we call *evidential reasoning*. The techniques are based upon the mathematics of belief functions developed by Dempster and Shafer and have been successfully applied to a variety of problems including computer vision, multisensor integration, and intelligence analysis.

We have developed both a formal basis and a framework for implementating automated reasoning systems based upon these techniques. Both the formal and practical approach can be divided into four parts: (1) specifying a set of distinct propositional spaces, (2) specifying the interrelationships among these spaces, (3) representing bodies of evidence as belief distributions, and (4) establishing paths for the bodies of evidence to move through these spaces by means of evidential operations, eventually converging on spaces where the target questions can be answered. These steps specify a means for arguing from multiple bodies of evidence toward a particular (probabilistic) conclusion. Argument construction is the process by which such evidential analyses are constructed and is the analogue of constructing proof trees in a logical context.

This technology features the ability to reason from uncertain, incomplete, and occasionally inaccurate information based upon seven evidential operations: fusion, discounting, translating, projection, summarization, interpretation, and gisting. These operation are theoretically sound but have intuitive appeal as well.

In implementing this formal approach, we have found that evidential arguments can be represented as graphs. To support the construction, modification, and interrogation of evidential arguments, we have developed Gister. Gister provides an interactive, menu-driven, graphical interface that allows these graphical structures to be easily manipulated.

Our goal is to provide effective automated aids to domain experts for argument construction. Gister represents our first attempt at such an aid.

Automating Argument Construction[*][†]

John D. Lowrance

Artificial Intelligence Center
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

1 Introduction

Over the past five years [GLF81,Low82,LG83b,LG83a,LSG86,LGS86,Wes88], the Artificial
Intelligence Center at SRI has been developing a new technology to address the problem of
automated information management within real-world contexts. The result of this work is a
body of techniques for automated reasoning from evidence that we call *evidential reasoning.*
The techniques are based upon the mathematics of belief functions developed by Dempster
and Shafer [Dem68,Sha76,Sha86] and have been successfully applied to a variety of problems
including computer vision, multisensor integration, and intelligence analysis.

We have developed both a formal basis and a framework for implementating automated
reasoning systems based upon these techniques. Both the formal and practical approach can
be divided into four parts: (1) specifying a set of distinct propositional spaces (i.e., *frames
of discernment*), each of which delimits a set of possible world situations; (2) specifying the
interrelationships among these propositional spaces (i.e., *compatibility relations* in a *gallery*);
(3) representing bodies of evidence as belief distributions over these propositional spaces
(i.e., *mass distributions*); and (4) establishing paths (i.e., *analyses*) for the bodies of evidence
to move through these propositional spaces by means of evidential operations, eventually
converging on spaces where the target questions can be answered. These steps specify
a means for arguing from multiple bodies of evidence toward a particular (probabilistic)
conclusion. Argument construction is the process by which such evidential analyses are
constructed and is the analogue of constructing proof trees in a logical context.

This technology features the ability to reason from uncertain, incomplete, and occasion-
ally inaccurate information (these being the characteristics of the information available in
real-world domains). It provides options for the representation of information: independent
opinions are expressed by multiple (independent) bodies of evidence; dependent opinions

opinions are expressed by multiple (independent) bodies of evidence; dependent opinions can be expressed either by a single body of evidence or by a network (i.e., analysis) that describes the interrelationships among several bodies of evidence. These networks of bodies of evidence capture the genealogy of each body (similar in spirit to [Coh85]) and are used in a manner similar to data-flow models [WA84] automatically updating interrelated beliefs (i.e., for belief revision [Doy81]). The technology includes the following evidential operations, which are based in theory but have intuitive appeal as well:

- FUSION—This operation pools multiple bodies of evidence into a single body of evidence that emphasizes points of agreement and deemphasizes points of disagreement.

- DISCOUNTING—This operation adjusts a body of evidence to reflect its source's credibility. If a source is completely reliable, discounting has no effect; if it is completely unreliable, discounting strips away all apparent information content; otherwise, discounting reduces the apparent information content in proportion to the source's unreliability.

- TRANSLATION—This operation moves a body of evidence away from its original context to a related one, to assess its impact on dependent hypotheses. For example, a body of evidence pertaining to the location of a ship can be translated to estimate its activity.

- PROJECTION—This operation moves a body of evidence away from its original temporal context, to a related one. For example, a report might make direct statements that pertain to a ship's location at a particular time. Through projection, this evidence can be used to estimate the possible locations of this ship at other times, either future or past.

- SUMMARIZATION—This operation eliminates extraneous details from a body of information. The resulting body of evidence is slightly less informative, but remains consistent with the original.

- INTERPRETATION—This operation calculates the "truthfulness" of a given statement based upon a given body of evidence. It produces an estimate of both the positive and negative effects of the evidence on the truthfulness of the statement.

- GISTING—This operation produces a single statement that captures the general sense of a body of evidence, without reporting degrees of uncertainty.

In implementing this formal approach, we have found that the gallery, frames, compatibility relations, and analyses can all be represented as graphs consisting of nodes connected by directed edges. To support the construction, modification, and interrogation of evidential structures, we have developed $\mathbf{Gister}^{TM}$. Gister provides an interactive, menu-driven, graphical interface that allows these graphical structures to be easily manipulated. The user simply makes menu selections to add an evidential operation to an analysis, to modify operation parameters (e.g., discount rates), or to change any portion of a gallery, including its frames and compatibility relations. In response, Gister automatically updates the analyses.

Unlike other expert systems, Gister is designed as a tool for the domain expert. With this tool, an expert can quickly and flexibly develop an argument (i.e., a line of reasoning) specific to a given domain situation. Gister helps the expert keep track of the complex inter-relationships among the components of his arguments, insure that the relevant information has been properly incorporated, and reveal the more tenative aspects of the arguments. This differs markedly from other expert systems where a single line of reasoning is developed by an expert and then is instantiated over different situations by nonexperts.

Our goal is to provide effective automated aids to domain experts for argument construction. Gister represents our first attempt at such an aid.

2 Background

Over the last five years, *expert-system technology* has emerged from the artificial intelligence (AI) research laboratories and has entered the market place. This technology has emerged in the form of *expert-system shells*, application-independent software systems that support the construction and automated use of knowledge bases. As a result, expert systems are being developed for a wide variety of applications.

Most of the commercially available shells use *production rules* as their formalism for representing knowledge. In its simplest form, the production rule (also termed an "if-then" rule) consists of an *antecedent* (i.e., the "if" part) and a *consequent* (i.e., the "then" part). The interpretation of the rule is: given information establishing the truth of the antecedent, the truth of the consequent can be inferred. In practice, a production rule is applied by attempting to match its antecedent against a database of facts; when a successful match is made, the consequent is added to the database.

However, expert knowledge is frequently suggestive rather than conclusive; the rules may therefore include a *strength* that is related to the conditional probabilities of the consequent given the antecedent and the antecedent given the consequent. Thus, based upon the current confidence in the antecedent in the database of facts and on the rule's stength, a confidence is derived for the consequent. This consequent may match the antecedent of other rules and thereby trigger their activation, thus resulting in the propagation of the influence of the original match throughout the database. Of course, multiple rules may share a common consequent, in which case it is necessary to have a means of resolving different confidence estimates that are derived through the use of different rules.

One way to visualize such a knowledge base is as a directed graph (Figure 1), where each antecedent and consequent is represented by a node and each rule is represented by a directed arc connecting its antecedent to its consequent. In essence, this *inference network* [DHN81] represents an argument, a line of reasoning that explains how certain premises support certain conclusions. Given probabilistic estimates of the truthfulness of certain facts, probabilistic conclusions can be automatically drawn. Thus, if the knowledge base (i.e., argument) was developed by an expert in the domain of application, his expertise becomes accessible to nonexperts.

In complex domains, it is frequently extremely difficult to select and coordinate all of the potentially relevant rules. The strength, and simultaneous weakness, of the rule-based
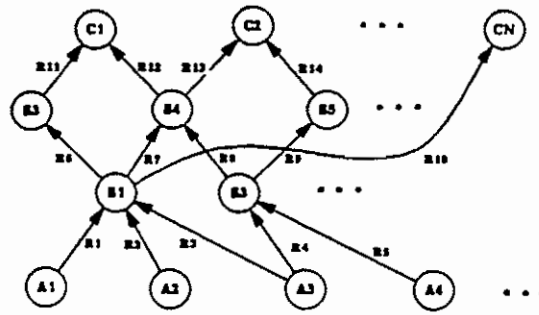
3

Figure 1: Inference Network of Production Rules.

approach is that the knowledge is represented in small chunks. In principle, each rule captures a very limited piece of knowledge (i.e., how one concept is directly related to another), and each is easily elicited from an expert and is easily understood. The original concept was that one could establish the validity of each rule in isolation and that the validity of the entire rule base would follow. Unfortunately, this is not the case. The confidences and strengths often do not represent sufficient probabilistic information to solve uniquely for the probabilities of the facts. In addition, the confidences and strengths represent subjective estimates, which are invariably inconsistent. In theory, a holistic approach, where more detailed information is required and inconsistencies are resolved during construction of the knowledge base, solves this problem, but in practice, such an approach makes construction of the knowledge base a monumental task. Therefore, heuristic methods for confidence propagation have been adopted.

The use of heuristic methods for uncertain reasoning in expert systems complicates construction of the knowledge base. [For example, one must determine what is to play the roles of antecedent and consequent. However, if one does not know ahead of time which of two events that tend to co-occur might be observed first, then one cannot establish which event should be the antecedent and which the consequent. One might suggest that two rules be included, with antecedent and consequent reversed, relative to each other. However, this solution is explicitly prohibited because of instability in the heuristic propagation methods.] In addition, since no true theoretical foundation exists, validity of a knowledge base can only be verified through experimental testing, often in the context of a set of hypothetical situations. The knowledge-base designer must coordinate the collective impact of the rules over the many long interacting chains of potential inferences, adding, modifying, and deleting rules, without benefit of any substantial structural guidelines.

Laboratory successes with this technology required close collaboration between application domain experts and AI researchers. The domain experts know their field, but not how to represent their knowledge in terms of a rule base; AI researchers understand how rules might be used to codify knowledge for automated reasoning, but lack expertise in the application domain. Each had to learn something of the other's field if a successful system was to be constructed. More recently the concept of a *knowledge engineer* has emerged as someone who specializes in eliciting and representing knowledge in terms of rules.

Despite the fact that most of the major laboratory successes in expert systems relied heavily upon uncertain reasoning (e.g., MYCIN [Sho76], an expert system for medical con-

4

sultation and PROSPECTOR [DHR77], a consultation system for mineral exploration), the major expert-system shells available in the market place today provide no true facility for uncertain reasoning. This deficiency can be directly traced to the difficulty of coping with uncertainty in rule-based systems.

Rule-based expert systems are applicable to those application domains where a single argument can be prespecified (including the relevant inputs, outputs, interrelationships, and flow of inferences), then instantiated over different situational data, without change to the argument, to solve the selected problem. Manipulation of the rule base is prohibited during its application to a specific situation, because of its fragility. Any change to match the current situation might adversely effect the validity of the knowledge base, thereby requiring that it be thoroughly retested before further application.

3   Requirements for Automated Argument Construction

Expert-system techniques are adequate for applications where the potential relevance of pertinent information can be prespecified. However, an important aspect of some application domains (e.g., intelligence analysis) is the ability to link what might first seem to be both irrelevant and unrelated pieces of information in unique ways that permit new conclusions to be drawn. This linking requires far more flexible interaction of the expert with the knowledge.

A system that can support automated argument construction must satisfy a number of requirements. First, its knowledge representations and operations, with which the user needs to interact, should be intuitive and easily understood. The user, who is interested in an answer to his problem and not in uncertain-reasoning techniques, should not be burdened with technical matters that are a function of the underlying technology. Although it is by no means a requirement, we have found that graphically oriented representations are often a particularly good choice.

Second, the knowledge should be represented in such a way as to be independent of how it is eventually used during argument construction. That is, the user should be able to state the facts directly as he sees them, without having to specify how they might be used in an argument. If two events tend to co-occur, then the user should be able to so state, without having to select one event as an indicator for the other.

Third, the knowledge must be easily modifiable. This requirement includes both the ability to make a modification quickly and the ability to understand its impact. Toward this end, the system should guide the user through any modification step by step, if multiple actions are required on the part of the user. Once a modification is complete, the argument should automatically react to the change, updating any conclusions that depend upon it. Thus, the system serves as an experimental environment in which a space of alternative formulations can be easily explored.

Fourth, the system's information requirements should match the availability and precision of the information in the problem domain. If prior probabilities cannot be reasonably assessed, they should not be required, or they should be representable in a form that retains their tentative nature.

5

Finally, the system needs to be theoretically well grounded. If it is not, both its stability and understandability suffer. If the user is to be free to explore his problem, the system must be flexible enough to support the user's exploration, without fear of its collapse because of heuristic frailties. In addition, there should be opportunity for true analysis, not just experimental testing. The user must be able to analyze his argument to understand its structure and its sensitivities.

Although satisfying these requirements will obviously be very difficult, we have taken some initial steps toward developing such automated aids for argument construction.

4   Constructing Arguments

Gister divides the problem of argument construction into two major steps: framing the problem and analyzing the evidence. In framing the problem, the user establishes a gallery of frames and compatibility relations that delimits a space of possibilities. In analyzing the evidence, each body of evidence is represented relative to a frame in the gallery and a sequence of evidential operations is established. This sequence determines how the evidence is transformed into pertinent conclusions. Collectively the gallery of frames and compatibility relations, together with the analyses, are the rough equivalent of an expert system's knowledge base.

4.1   Framing the Problem

The first step in applying evidential reasoning to a given problem is to delimit a propositional space of possible situations, exactly one of which is true at any one time. Within the theory of belief functions, this propositional space is called the *frame of discernment*. If the problem to be addressed were that of locating a ship, then the frame of discernment would consist of the set of all possible locations for that vessel. This frame might be represented by a set Θ_A, in which each element a_i corresponds to a possible location:

$$\Theta_A = \{a_1, a_2, \ldots, a_n\} \quad .$$

Once a frame of discernment has been established, propositional statements can be represented by disjunctions of elements from the frame corresponding to those situations for which the statements are true. For example, the proposition A_i might correspond to the statement that the vessel is docked, in which case A_i would be represented by the subset of elements from Θ_A that correspond to possible locations adjacent to docks:

$$A_i \subseteq \Theta_A \quad .$$

In implementing this formal approach, we have found that frames, like the other formal elements in this theory, can be straightforwardly represented as graphs consisting of nodes connected by directed edges. Because they are graphs, these formal elements are easily understood, and they provide an intuitive basis for man-machine interaction. A frame is
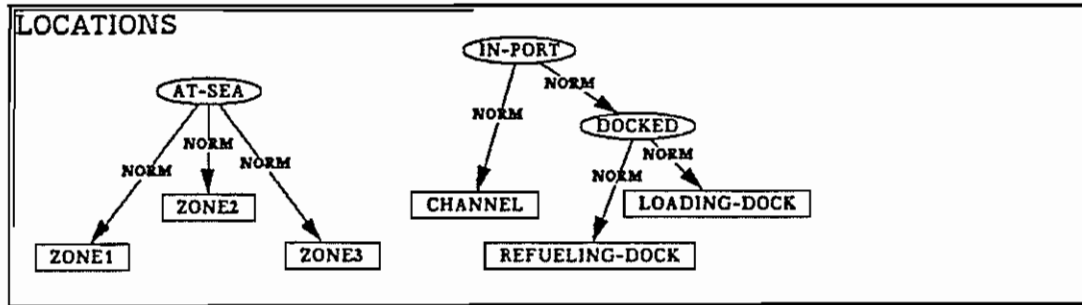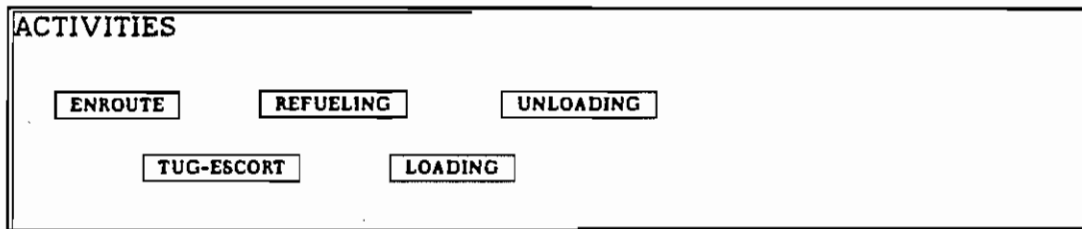
Figure 2: LOCATIONS Frame.

Figure 3: ACTIVITIES Frame.

represented by a named graph that includes a node for each element of the frame and may include additional nodes representing aliases, i.e., named disjunctions of elements. Each of these additional nodes has edges pointing to elements of the frame (or other aliases) that make up the disjunction. Here, the possible locations for a ship might be represented by a graph named LOCATIONS (Figure 2)[1] that includes six elements (ZONE1, ZONE2, ZONE3, CHANNEL, LOADING-DOCK, REFUELING-DOCK) and three aliases (IN-PORT, DOCKED, AT-SEA).

If other aspects of ships are of interest besides their location, then additional frames of discernment might be defined. For example, the activities of these ships might be of interest. If so, an additional frame Θ_B might be defined to include elements corresponding to refueling, loading cargo, unloading cargo, being enroute, and being under tug escort. Propositional statements pertaining to a ship's activity can then be defined relative to this frame; e.g.,

$$\Theta_B = \{b_1, b_2, \ldots, b_n\}$$
$$B_j \subseteq \Theta_B .$$

The frame is represented by a graph named ACTIVITIES (Figure 3) that includes five elements: ENROUTE, TUG-ESCORT, UNLOADING, LOADING, REFUELING.

---

[1]All of the figures in this section are actual screen images from Gister.

7

So far, propositional statements pertaining to a ship's location or to its activity can be addressed separately, but they cannot be jointly considered. To do this, one must first define a *compatibility relation* between the two frames. A compatibility relation simply describes which elements from the two frames can be true simultaneously. For example, a ship located at a loading dock might be loading or unloading cargo, but is not refueling, or enroute. In other words, being located at a loading dock is compatible only with one of two activities, loading or unloading. Thus, the compatibility relation between frames Θ_A and Θ_B is a subset of the cross product of the two frames. A pair (a_i, b_j) is included if and only if they can be true simultaneously. There is at least one pair (a_i, b_j) included for each a_i in Θ_A (the analogue is true for each b_j):

$$\Theta_{A,B} \subseteq \Theta_A \times \Theta_B \quad .$$

Using the compatibility relation $\Theta_{A,B}$, we can define a *compatibility mapping* $C_{A \rightarrow B}$ for translating propositional statements expressed relative to Θ_A to statements relative to Θ_B. If a statement A_k is true, then the statement $C_{A \rightarrow B}(A_k)$ is also true:

$$
\begin{aligned}
C_{A \rightarrow B} &: 2^{\Theta_A} \mapsto 2^{\Theta_B} \\
C_{A \rightarrow B}(A_k) &= \{b_j | (a_i, b_j) \in \Theta_{A,B}, a_i \in A_k\} \quad .
\end{aligned}
$$

A compatibility relation is represented as a graph that includes the nodes from the frames that it relates with edges connecting compatible elements. For example, in the LOCATIONS-ACTIVITIES compatibility relation (Figure 4) relating the LOCATIONS and ACTIVITIES frames, ZONE1, ZONE2, and ZONE3 are all connected to ENROUTE (becuase these zones represent areas at sea), CHANNEL is connected to TUG-ESCORT (because a ship entering or leaving the port at the end of this channel would be under tugboat control), LOADING-DOCK is connected to both LOADING and UNLOADING (because either activity is consistent with being at that dock), and REFUELING-DOCK is connected to REFUELING. The directed edges define the compatibility mapping from LOCATIONS to ACTIVITIES, moving forward along the edges, and the mapping from ACTIVITIES to LOCATIONS, moving backward along the edges.

Instead of translating propositional statements between these two frames via $C_{A \rightarrow B}$ and $C_{B \rightarrow A}$, we might choose to translate these statements to a common frame that captures all of the information. This common frame is identical to the compatibility relation $\Theta_{A,B}$. Frame Θ_A (and analogously Θ_B) is trivially related to frame $\Theta_{A,B}$ via the following compatibility relation and compatibility mappings:

$$
\begin{aligned}
\Theta_{A,(A,B)} &= \{(a_i, (a_i, b_j)) | (a_i, b_j) \in \Theta_{A,B}\} \\
C_{A \rightarrow (A,B)}(A_k) &= \{(a_i, b_j) | (a_i, (a_i, b_j)) \in \Theta_{A,(A,B)}, a_i \in A_k\} \\
&= \{(a_i, b_j) | (a_i, b_j) \in \Theta_{A,B}, a_i \in A_k\} \\
C_{(A,B) \rightarrow A}(X_k) &= \{a_i | (a_i, b_j) \in \Theta_{A,B}, (a_i, b_j) \in X_k\} \quad .
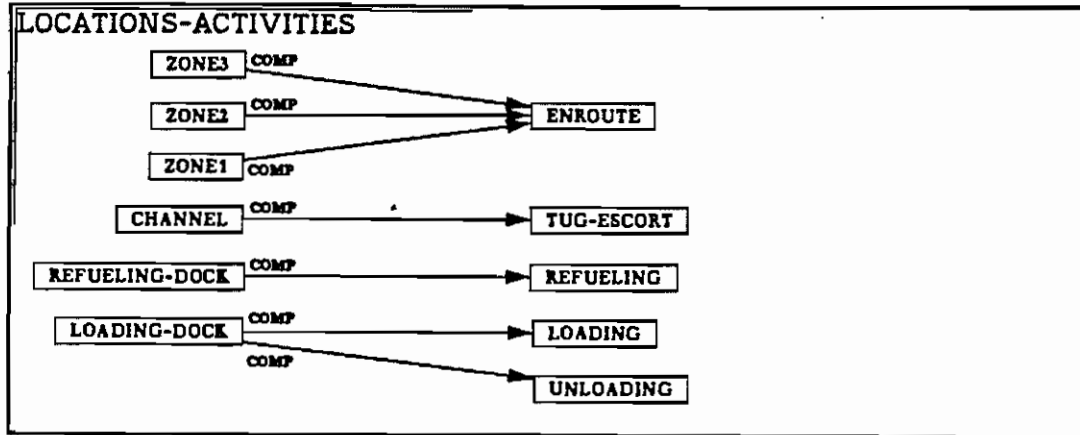\end{aligned}
$$

8

Figure 4: LOCATIONS-ACTIVITIES Compatibility Relation.

Clearly, as more aspects of these ships become of interest, the number and complexity of the frames and compatibility mappings increases. However, there is a trade-off between the complexity of individual frames and the complexity of the network of compatibility mappings connecting them. We might define a single (complex) frame that encompasses all aspects of interest or, alternatively, define a (complex) network of frames that includes a distinct frame for each aspect of interest. In fact, Gister provides facilities for generating composite frames from multiple interrelated frames.

Of course, a network of interrelated frames and a single (complex) frame may not be equivalent. For example, consider the following frame:

$$\Theta_{A,B,C} = \{(a_1, b_1, c_1), (a_2, b_1, c_2), (a_2, b_2, c_2)\} \quad .$$

If this frame properly captures the relationship among frames Θ_A, Θ_B, and Θ_C, then c_1 is the only element from Θ_C compatible with a_1 from Θ_A. However, if we maintain these as three separate frames connected by compatibility mappings, $C_{A \mapsto B}, C_{B \mapsto A}, C_{B \mapsto C}$, and $C_{C \mapsto B}$, both c_1 and c_2 are compatible with a_1 because a_1 is compatible with b_1, and b_1 is compatible with both c_1 and c_2; i.e., $C_{B \mapsto C}(C_{A \mapsto B}(\{a_1\})) = \{c_1, c_2\}$. However, if a_1 is true, then it follows that either c_1 or c_2 is true. Thus, the reasoning based on a well-formed *gallery* of interconnected frames is sound but not necessarily complete. A gallery is well formed if there exists a single all-encompassing frame whose answers are always included in the answers based upon the gallery.

In dynamic environments, compatibility relations can be used to reason over time. If Θ_{A1} represents the possible states of the world at time one and Θ_{A2} represents the possible states at time two, then a compatibility relation, $\Theta_{A1,A2}$, can capture the possible state transitions. For example, if Θ_{A1} and Θ_{A2} both represent the possible locations of a ship (i.e., they are identical to Θ_A as previously defined), then $\Theta_{A1,A2}$ could represent the constraints on that ship's movement. A pair of locations (a_i, a_j) would be included in $\Theta_{A1,A2}$ if a ship located at a_i on Day 1 (i.e., time) could reach a_j by Day 2. If we assume that the possible movements of a ship are constrained in the same way over any two-day period, then the
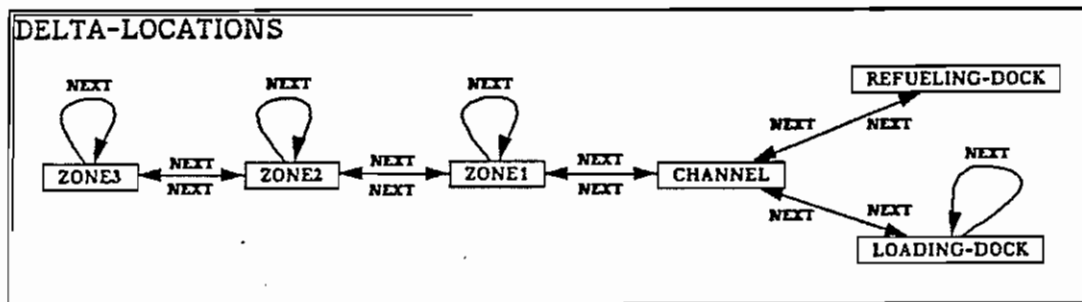
9

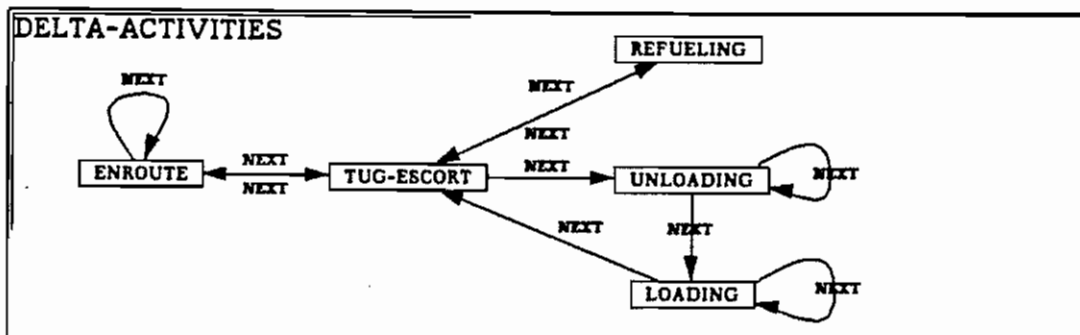Figure 5: DELTA-LOCATIONS Compatibility Relation.

Figure 6: DELTA-ACTIVITIES Compatibility Relation.

compatibility mapping associated with this compatibility relation can be reapplied as many times as necessary to constrain the possible locations of a ship across an arbitrary number of days.

DELTA-LOCATIONS and DELTA-ACTIVITIES (Figures 5 and 6) are two compatibility relations that relate frames to themselves. They represent possible state transitions in their respective frames over any two day period. Edges connect compatible elements from one day to the next. DELTA-LOCATIONS indicates that the zones are linearly ordered and that a ship must pass through the channel to get to either the loading or refueling docks. It also indicates that a ship will remain at the refueling dock or in the channel only for one day at a time but may remain anywhere else for any number of days. In DELTA-ACTIVITES it can be seen that a ship must progress through TUG-ESCORT from ENROUTE before proceeding to REFUELING or UNLOADING and that REFUELING and TUG-ESCORT are one-day activities. Further, a ship must go through LOADING after UNLOADING before returning to TUG-ESCORT.

Finally, the overall topology of the gallery is represented as a graph. Here (Figure 7), the LOCATIONS and ACTIVITIES frames are represented as nodes, and the three compatibility relations are represented as edges. LOCATIONS-ACTIVITIES, the compatibility relation that relates the LOCATIONS frame to the ACTIVITIES frame, is represented by an edge from LOCATIONS to ACTIVITIES. The other two compatibility relations, DELTA-LOCATIONS and DELTA-ACTIVITIES, relate frames to themselves and are represented by edges that begin and end at the same node.
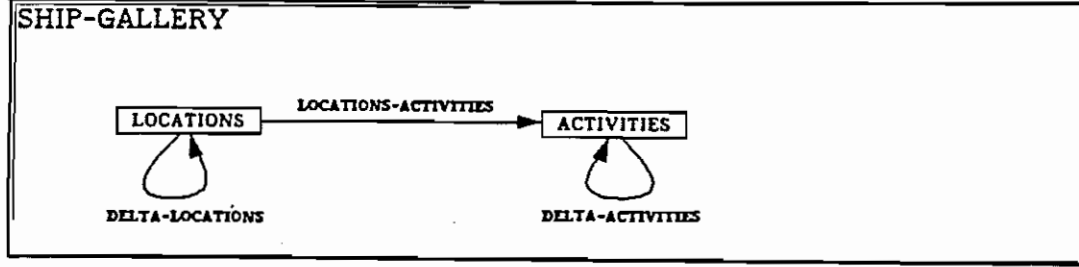
10

Figure 7: SHIP-GALLERY Gallery.

4.2 Analyzing the Evidence

Once a gallery has been established, Gister can analyze the available evidence. The goal of this analysis is to establish a line of reasoning, based upon both the possibilistic information in the gallery and the probabilistic information from the evidence, that determines the most likely answers to some questions. The gallery delimits the space of possible situations, and the evidential information establishes the likelihoods of these possibilities. Within an analysis, bodies of evidence are expressed relative to frames in the gallery, and paths are established for the bodies of evidence to move through the frames via the compatibility mappings. An analysis also specifies if other evidential operations are to be performed, including whether multiple bodies of evidence are to be combined when they arrive at common frames. Finally, an analysis specifies which frame and ultimate bodies of evidence are to be used to answer each target question. Thus, an analysis specifies a means for arguing from multiple bodies of evidence toward a particular (probabilistic) conclusion. An analysis, in an evidential context, is the analogue of a proof tree in a logical context.

To begin, each body of evidence is expressed relative to a frame in the gallery. Each is represented as a mass distribution (e.g., m_A) over propositional statements discerned by a frame (e.g., Θ_A):

$$
\begin{aligned}
m_A : 2^{\Theta_A} &\mapsto [0,1] \\
\sum_{A_i \subseteq \Theta_A} m_A(A_i) &= 1 \\
m_A(\emptyset) &= 0 .
\end{aligned}
$$

Intuitively, mass is attributed to the most precise propositions a body of evidence supports. If a portion of mass is attributed to a proposition A_i, it represents a minimal commitment to that proposition and all the propositions it implies. Additional mass attributed to a proposition A_j that is compatible with A_i, but does not imply it (i.e., $\emptyset \neq A_i \cap A_j \neq A_j$), represents a potential commitment: mass that neither supports nor denies that proposition at present but that might later move either way based upon additional information.

To *interpret* this body of evidence relative to the question A_j, we calculate its *support* and *plausibility* to derive its *evidential interval* as follows:

11

$$Spt(\mathsf{A}_j) = \sum_{\mathsf{A}_i \subseteq \mathsf{A}_j} m_A(\mathsf{A}_i)$$

$$Pls(\mathsf{A}_j) = 1 - Spt(\Theta_A - \mathsf{A}_j)$$

$$[Spt(\mathsf{A}_j), Pls(\mathsf{A}_j)] \subseteq [0,1] \ .$$

The lower bound of an evidential interval indicates the degree to which the evidence supports the proposition, while the upper bound indicates the degree to which the evidence fails to refute the proposition, i.e., the degree to which it remains plausible.

Discounting is an evidential operation that adjusts a mass distribution to reflect its source's credibility (expressed as a discount rate $r \in [0,1]$). If a source is completely reliable ($r = 0$), discounting has no effect; if it is completely unreliable ($r = 1$), discounting strips away all apparent information content; otherwise, discounting lowers the apparent information content in proportion to the source's unreliability. It has the effect of widening the evidential intervals, reflecting increased ignorance. Discounting is defined as follows:

$$m_A^{\%}(\mathsf{A}_i) = \begin{cases} (1-r)m_A(\mathsf{A}_i), & \mathsf{A}_i \neq \Theta_A \\ r + (1-r)m_A(\Theta_A), & \text{otherwise} \end{cases} \ .$$

If a body of evidence is to be interpreted relative to a question expressed over a frame different from the one over which the evidence is expressed, a path of compatibility relations connecting the two frames is required. The mass distribution expressing the body of evidence is then repeatedly *translated* from frame to frame, by way of compatibility mappings, until it reaches the ultimate frame of the question. In translating m_A from frame Θ_A to frame Θ_B by way of compatibility mapping $C_{A \mapsto B}$, the following computation is applied to derive the translated mass distribution m_B:

$$m_B(\mathsf{B}_j) = \sum_{C_{A \mapsto B}(\mathsf{A}_i) = \mathsf{B}_j} m_A(\mathsf{A}_i) \ .$$

Intuitively, if we (partially) believe A_i, and A_i implies B_j, then we should have the same (partial) belief in B_j. This method is also applied to move mass distributions among frames that represent states of the world at different times; however, when this is the case, the operation is called *projection*.

Once two mass distributions m_A^1 and m_A^2 representing independent opinions are expressed relative to the same frame of discernment, they can be *fused* (i.e., combined) using *Dempster's Rule of Combination*. Dempster's rule pools mass distributions to produce a new mass distribution m_A^3 that represents the consensus of the original disparate opinions. That is, Dempster's rule produces a new mass distribution that leans toward points of agreement between the original opinions and away from points of disagreement. Dempster's rule is defined as follows:

$$m_A^3(A_k) = (1-k)^{-1} \sum_{A_i \cap A_j = A_k} m_A^1(A_i) m_A^2(A_j)$$

$$k = \sum_{A_i \cap A_j = \emptyset} m_A^1(A_i) m_A^2(A_j) \neq 1 \quad .$$

Because Dempster's rule is both commutative and associative, multiple (independent) bodies of evidence can be combined in any order without affecting the result. If the initial bodies of evidence are independent, then the derivative bodies of evidence are independent as long as they share no common ancestors. Thus, in the course of constructing an analysis, we must take care that evidence is propagated and combined in such a way as to guarantee the independence of the evidence at each combination. Gister protects the user by tracking the evidence and preventing such dependent combinations.

Another evidential operation is *summarization*. Summarization eliminates extraneous details from a mass distribution by collecting all of the extremely small amounts of mass (determined by a threshold $t \in [0,1]$) attributed to propositions and attributing the sum to the disjunction of those propositions. The resulting mass distribution is slightly less informative than the original (i.e., some evidential intervals based upon this resulting mass distribution will be wider than those based upon the original), but it remains consistent with the original (i.e., the intervals based on the resulting distribution contain those based on the original):

$$m_A^+(A_i) = \begin{cases} m_A(A_i), & A_i \neq S \\ s + m_A(S), & \text{otherwise} \end{cases}$$

$$S = \bigcup_{0 \neq m_A(A_i) < t} A_i$$

$$s = \sum_{0 \neq m_A(A_i) < t} m_A(A_i) \quad .$$

Another form of summarization can be defined in terms of translation. If mass is distributed over a fairly fine-grained frame (i.e., a frame with a large number of elements because it preserves subtle distinctions), but the question at hand could be resolved in the context of a coarser frame (i.e., one with fewer elements that makes fewer distinctions) that is related to the finer by means of a compatibility relation, then the mass distribution can be translated to the coarser frame to reduce its complexity. Like the thresholded summarization operation defined above, the resulting mass distribution is generally less informative but consistent, so long as the gallery is well formed. Thus, the first summarization operation defined above discards information that will have a miniscule impact on any questions, while the second summarization technique discards details that are irrelevant to a particular set of questions (i.e., the set of propositions discerned by the frame).

Gisting is another evidential operation that can be included in an analysis. Gisting produces a (Boolean-valued) statement that attempts to capture the essence of a mass

13

distribution. In other words, it attempts to summarize the contents of a body of evidence in terms of a single statement from the frame, void of any uncertainty or ignorance. Such a summary is particularly useful when explaining lines of reasoning. As defined below, the gist of a mass distribution is the most pointed statement from the frame whose support meets or exceeds a selected level. The gist[2,3], G, of a mass distribution, m_A, is defined relative to a gist level, $g \in [0, 1]$:

$$G = \bigcup_{A_i \in G} A_i$$

$$G \subseteq 2^{\Theta_A}$$

for all $A_i, A_j \in G, A_k \notin G$

$$Spt(A_i) = Spt(A_j) \geq g$$
$$|A_i| = |A_j|$$
$$Spt(A_i) > Spt(A_k) \quad or \quad |A_k| > |A_i| \quad .$$

Still other evidential operations allow an established analysis to be examined and revised. These operations are not included in analyses, but are used as tools to assist in their development.

After the gallery and its supporting frames and compatibility relations have been established, evidential analyses can be constructed. These *analyses* are represented as data-flow graphs where the data and the operations are evidential. Figure 8 is one such analysis. Here primitive bodies of evidence are represented by elliptical nodes, and derivative bodies of evidence are represented by circular nodes. Diamond-shaped nodes represent interpretations of bodies of evidence. The values of these nodes are used as repositories for the information (i.e., data) that they represent (Figure 9). For bodies of evidence, this information includes a frame of discernment (including the day to which the evidence pertains), a mass distribution, and other supporting information. Edges pointing to a derivative node are labeled with the evidential operation that is applied to the bodies of evidence, at the other ends of the edges, to derive the body of evidence represented by this node.

Figure 8 includes the menus for working with analyses. On the left side of the screen is a menu of nouns. The user determines with what class of objects he wishes to work and selects the appropriate noun from the menu. Once a noun has been selected, a menu of verbs appears on the right side of the screen. A selection from this menu invokes the operation corresponding to the selected verb on the previously selected noun. The user then designates the appropriate nodes, edges, and the like for the selected operation. Gister provides a similar set of menus for interacting with the gallery.

---

[2]This definition uses cardinality as a measure of specificity (i.e., pointedness) and thereby assumes that all elements of Θ_A are equally specific. It also ignores problems of instability stemming from small variations in the support, specificity, and the gist level.

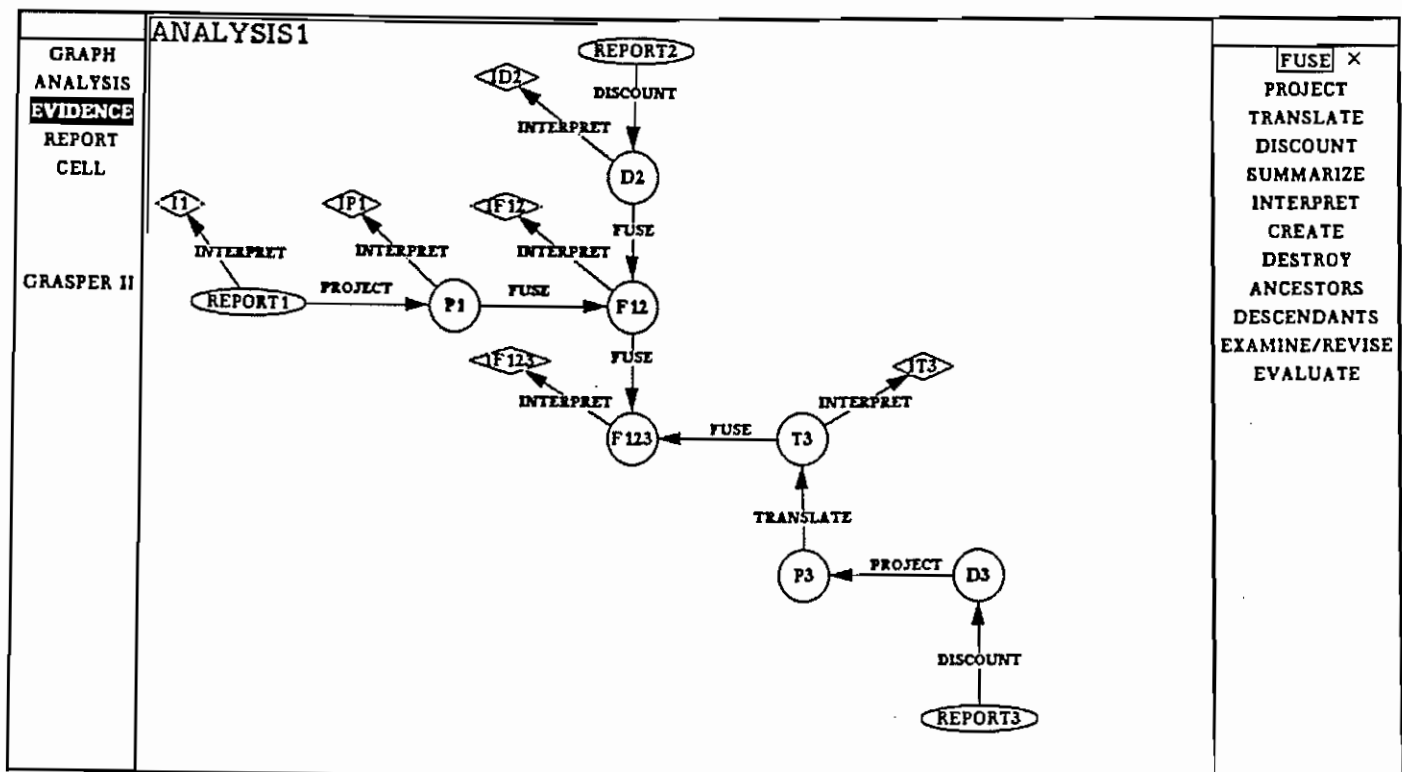[3]Other definitions are under investigation.

Figure 8: ANALYSIS1 Analysis.

```
REPORT1
TYPE: EVIDENCE
FOD: (LOCATIONS 1.)
MASSFUN: (((CHANNEL) 0.7) ((ZONE1) 0.3))
Exit ▢
```

```
P1
TYPE: PROJECTION
DELTA-T: 1.
FOD: (LOCATIONS 2.)
MASSFUN: (((REFUELING-DOCK LOADING-DOCK ZONE1) 0.7) ((ZONE2 CHANNEL ZONE1) 0.3))
Exit ▢
```

```
REPORT2
TYPE: EVIDENCE
FOD: (LOCATIONS 2.)
MASSFUN: (((CHANNEL LOADING-DOCK REFUELING-DOCK) 1.0))
Exit ▢
```

```
D2
TYPE: DISCOUNT
DISCOUNT-RATE: 20.
FOD: (LOCATIONS 2.)
MASSFUN: (((CHANNEL LOADING-DOCK REFUELING-DOCK) 0.8) ((REFUELING-DOCK ZONE2 CHANNEL LOADING-DOCK ZONE1 ZONE3) 0.2))
Exit ▢
```

```
F12
TYPE: FUSION
FOD: (LOCATIONS 2.)
MASSFUN: ((LOADING-DOCK REFUELING-DOCK) 0.56)
        ((CHANNEL) 0.24000001)
        ((REFUELING-DOCK LOADING-DOCK ZONE1) 0.14)
        ((ZONE2 CHANNEL ZONE1) 0.060000002))
CONFLICT: 0.0
Exit ▢
```

```
REPORT3
TYPE: EVIDENCE
FOD: (ACTIVITIES 3.)
MASSFUN: (((LOADING) 1.0))
Exit ▢
```

```
D3
TYPE: DISCOUNT
DISCOUNT-RATE: 40.
FOD: (ACTIVITIES 3.)
MASSFUN: (((LOADING) 0.6) ((TUG-ESCORT UNLOADING ENROUTE LOADING REFUELING) 0.4))
Exit ▢
```

```
P3
TYPE: PROJECTION
DELTA-T: -1.
FOD: (ACTIVITIES 2.)
MASSFUN: (((UNLOADING LOADING) 0.6) ((REFUELING LOADING ENROUTE UNLOADING TUG-ESCORT) 0.4))
Exit ▢
```

```
T3
TYPE: TRANSLATION
THETA: LOCATIONS
FOD: (LOCATIONS 2.)
MASSFUN: (((LOADING-DOCK) 0.6) ((ZONE3 ZONE2 ZONE1 LOADING-DOCK REFUELING-DOCK CHANNEL) 0.4))
Exit ▢
```

```
F123
TYPE: FUSION
FOD: (LOCATIONS 2.)
MASSFUN:((LOADING-DOCK) 0.5121951)
        ((LOADING-DOCK REFUELING-DOCK) 0.2731707)
        ((CHANNEL) 0.11707317)
        ((ZONE1 LOADING-DOCK REFUELING-DOCK) 0.06829268)
        ((ZONE2 ZONE1 CHANNEL) 0.029268293))
CONFLICT: 0.17999995
Exit ▢
```

Figure 9: Data from ANALYSIS1.

16

In the analysis of a ship in Figures 8 and 9, there are three primitive bodies of evidence. REPORT1 locates the ship on Day 1, saying that there is a 70 percent chance that it can be found in the CHANNEL and a 30 percent chance that it is in ZONE1; REPORT2 says that the ship was IN-PORT on Day 2; and REPORT3 indicates that the ship was LOADING cargo on Day 3. REPORT1 is taken at face value, but REPORT2 and REPORT3 have been discounted by 20 percent and 40 percent, respectively, to derive D2 and D3, reflecting doubt in the credibility of these reports. REPORT1 has been projected forward by one day to derive P1. D3 has been projected backward in time by one day to derive P3 and then has been translated from the ACTIVITIES frame to the LOCATIONS frame. Finally, this result, T3, has been fused with F12 to derive a consensus, based on all three reports, about the ship's location on Day 2.

The interpretation nodes in this analysis track the evidential intervals for some key propositions. I1 is based solely on REPORT1 and indicates that there is precisely a 70 percent chance of the ship being IN-PORT[0.7, 0.7] and no chance of it being DOCKED [0.0, 0.0]on Day 1. IP1 indicates that, based solely upon REPORT1, after one day has elapsed, nothing is known about whether the ship is IN-PORT [0.0, 1.0], but that it may now be DOCKED [0.0, .7.0]. If REPORT2 is included after being discounted, IF12 indicates that there is strong reason to believe that the ship is IN-PORT [0.8, 1.0], but there is conflicting information concerning whether or not it is DOCKED [0.56, 0.7]. IT3 indicates that, based solely upon REPORT3, after having been discounted, projected backward a day, and translated to the LOCATION frame, that there is 0.6 support and 1.0 plausibility for both IN-PORT and DOCKED. Finally, when all three reports are considered, IF123 indicates strong belief that the ship is IN-PORT [0.9, 1.0] on Day 2 and a reasonably strong belief, though mixed, that it is also DOCKED [0.78, 0.85].

4.3 Exploring Alternative Arguments

Of course, the example above is not the only argument that can be constructed from these data. For example, the credibility given to the initial reports might be assessed differently. To explore such alternatives using Gister, the user has only to modify the discount rates stored on the appropriate discount nodes in ANALYSIS1. In response, Gister recalculates the dependent conclusions. Alternatively, the user might decide to develop a parallel line of reasoning within ANALYSIS1, constructing a new sequence of evidential operations with different parameters to argue for the same or a different conclusion. Or a completely new analysis under a different name might be constructed.

In addition, variations in the gallery can be explored. Here, elements might be deleted or new elements added to frames and compatibility relations to determine their effect on established analyses. Alternatively, different frames and compatibility relations might be constructed, after which new analyses might be created or old analyses modified to examine the data relative to a set of different possibilistic assumptions.

Gister separates the specific data about the current situation, from the general knowledge of what is possible, from the way in which this data and knowledge are utilized to draw conclusions. Further, it allows these to be independently varied, giving the user the

freedom to create and examine alternative formulations. Through this interaction, the user comes to better understand the basis and sensitivities of his arguments and conclusions.

In essence, our approach reflects the view that there is not so much a correct argument, but rather a set of alternative competing arguments that must be explored.

5 Summary

We have develped a system, Gister, that supports the construction, modification, and analysis of evidential arguments. Gister supports an interactive, menu-driven, graphical interface that allows these structures to be easily understood and manipulated. The user simply selects from a menu to add an evidential operation to an analysis, to modify operation parameters, or to change any portion of a gallery. In response, Gister updates the analyses, allowing the user to explore quickly the space of alternative arguments.

Unlike other expert systems, Gister is designed as a tool for the domain expert. With this tool, an expert can quickly and flexibly develop a line of reasoning specific to a given domain situation. At SRI, this approach has been applied to naval intelligence problems. New work is focusing on adapting this technology to multisource data fusion for the Army.

References

[Coh85] Paul Cohen. *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach.* Pitman Publishing, Inc., 1985.

[Dem68] Arthur P. Dempster. A generalization of Bayesian inference. *Journal of the Royal Statistical Society,* 30:205–247, 1968.

[DHN81] Richard O. Duda, Peter E. Hart, and Nils J. Nilsson. Subjective Bayesian methods for rule-based inference systems. In Bonnie Lynn Webber and Nils J. Nilsson, editors, *Readings in Artificial Intelligence,* pages 192–199, Tioga Publishing Company, Palo Alto, California, 1981.

[DHR77] Richard O. Duda, Peter E. Hart, and Rene Reboh. A rule-based consultation program for mineral exploration. In *Proceedings of the Lawrence Symposium on Systems and Decision Sciences,* pages 306–309, Berkeley, California, October 1977.

[Doy81] Jon Doyle. A truth maintenance system. In Bonnie Lynn Webber and Nils J. Nilsson, editors, *Readings in Artificial Intelligence,* pages 496–516, Tioga Publishing Company, Palo Alto, California, 1981.

[GLF81] Thomas D. Garvey, John D. Lowrance, and Martin A. Fischler. An inference technique for integrating knowledge from disparate sources. In *Proceeding of the Seventh Joint Conference on Artificial Intelligence,* pages 319–325, American Association for Artificial Intelligence, 445 Burgess Drive, Menlo Park, California, August 1981.

[LG83a] John D. Lowrance and Thomas D. Garvey. *Evidential Reasoning: An Approach to the Simulation of a Weapons Operation Center.* Technical Report, Artificial Intelligence Center, SRI International, 333 Ravenswood Avenue, Menlo Park, California, September 1983.

[LG83b] John D. Lowrance and Thomas D. Garvey. *Evidential Reasoning: An Implementation for Multisensor Integration.* Technical Report 307, Artificial Intelligence Center, SRI International, 333 Ravenswood Avenue, Menlo Park, California, December 1983.

[LGS86] John D. Lowrance, Thomas D. Garvey, and Thomas M. Strat. A framework for evidential-reasoning systems. In *Proceeding of the National Conference on Artificial Intelligence*, pages 896–903, American Association for Artificial Intelligence, 445 Burgess Drive, Menlo Park, California, August 1986.

[Low82] John D. Lowrance. *Dependency-Graph Models of Evidential Support.* PhD thesis, Department of Computer and Information Science, University of Massachusetts, Amherst, Massachusetts, September 1982.

[LSG86] John D. Lowrance, Thomas M. Strat, and Thomas D. Garvey. *Application of Artificial Intelligence Techniques to Naval Intelligence Analysis.* Final Report SRI Contract 6486, Artificial Intelligence Center, SRI International, 333 Ravenswood Avenue, Menlo Park, California, June 1986.

[Sha76] Glenn Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, Princeton, New Jersey, 1976.

[Sha86] Glenn Shafer. Belief functions and possibility measures. *The Analysis of Fuzzy Information*, 1, 1986. In press.

[Sho76] Edward E. Shortliffe. *Computer-Based Medical Consultations: MYCIN.* Elsevier, New York, New York, 1976.

[WA84] W. W. Wadge and E. A. Ashcroft. *Lucid, The Dataflow Programming Language.* Academic Press U. K., 1984.

[Wes86] Leonard P. Wesley. *Evidential-Based Control in Knowledge-Based Systems.* PhD thesis, Department of Computer and Information Science, University of Massachusetts, Amherst, Massachusetts, 1986. In preparation.